IDR: a participatory methodology for interdisciplinary design in technology enhanced learning

Journal Item

IDR: a participatory methodology for interdisciplinary design in technology enhanced learning

Niall Winters[1] and Yishay Mor
London Knowledge Lab
Institute of Education, University of London
23-29 Emerald Street
London WC1N 3QS
UK

{n.winters, y.mor}@ioe.ac.uk

Abstract

One of the important themes that emerged from the CAL'07 conference was the failure of technology to bring about the expected disruptive effect to learning and teaching. We identify one of the causes as an inherent weakness in prevalent development methodologies. While the problem of designing technology for learning is irreducibly multi-dimensional, design processes often lack true interdisciplinarity. To address this problem we present IDR, a participatory methodology for interdisciplinary techno-pedagogical design, drawing on the design patterns tradition (Alexander, Silverstein & Ishikawa, 1977) and the design research paradigm (DiSessa & Cobb, 2004). We discuss the iterative development and use of our methodology by a pan-European project team of educational researchers, software developers and teachers. We reflect on our experiences of the participatory nature of pattern design and discuss how, as a distributed team, we developed a set of over 120 design patterns, created using our freely available open source web toolkit. Furthermore, we detail how our methodology is applicable to the wider community through a workshop model, which has been run and iteratively refined at five major international conferences, involving over 200 participants.

Keywords: Design; Methodology; Interdisciplinary projects; Design Patterns; Cooperative/collaborative learning

---

[1] Corresponding author

1 Introduction

A central theme of the CAL'07 conference was that technology enhanced learning (TEL) has not had the desired disruptive effect in the classroom. While the reasons for this are varied, we postulate that one cause may be because there is an inherent methodological weakness in the development of TEL environments. TEL as a field is characterised by interdisciplinarity. This is defined as follows (Committee on Science, Engineering and Public Policy, 2004):

> Interdisciplinary research is a mode of research by teams or individuals that integrates information, data, techniques, tools, perspectives, concepts, and/or theories from two or more disciplines or bodies of specialized knowledge to advance fundamental understanding or to solve problems whose solutions are beyond the scope of a single discipline or field of research practice.

Yet it seems much of the development of technical tools and pedagogical activities is multidisciplinary in nature, by definition a less integrated process (see Section 1.2). This is not to apportion blame but is a reflection on the pragmatic constraints that TEL researchers and practitioners have been presented with. Each community brings to the field their own practices and experience; software developers rely on well-trialled engineering principles when building software; teachers are well versed in course and activity design and so on. Perhaps because of this, TEL lacks the clear theoretical and methodological foundations that have long existed in the nature sciences.

To address this significant issue within TEL, we present the IDR methodology for interdisciplinary design. The methodology was developed during the 'learning patterns' project. This was a 1-year project involving partner institutions across six European countries with expertise in computer science, educational technology, teaching, pedagogical design and games. The network further involved partner schools in three of the six countries, with 21 people making up the core of the team. The main aim of the project was to identify, elaborate and connect *design knowledge* from the various domains of expertise within and across the project. IDR was the methodology developed and used. The primary output of IDR was a set of over 120 design patterns (web link to pattern database removed for blind review), explicitly detailing our interdisciplinary practices over the course of the project. To use the methodology effectively, we developed an associated web toolkit (see Section 3). As detailed in Section 4.2, these were further validated and iteratively refined at five international conferences by participants from both academia and industry.

The need to accumulate and reuse design knowledge is recognised as a major challenge for the TEL community. McAndrew et al (2006) review several widely acknowledged frameworks which claim to address this challenge. Existing learning design frameworks tend to be very concrete. Thus, they are powerful for rapid production of quality materials once the design is specified, but weak in supporting a higher-level discussion. To address this issue, our methodology is underpinned by a design patterns approach (Alexander, Silverstein & Ishikawa, 1977). Design patterns provide a means for sharing abstractions of methods for solving design problems. In this work, we extend this view and distinguish our approach by the consideration that anyone involved in the production or use of a technology is an expert in one facet of techno-pedagogical design. We therefore postulate that an effective design process needs to bring in the expertise of *all* those involved, and thus requires a rich yet accessible design language. Furthermore, if we wish to improve the process of design we need to look at its full cycle, and not just its software outputs (see Section 1.4). Thus, participation does not solely rely on the development of software applications but can be evidenced by other outputs including, for example, design patterns and pedagogical plans. The IDR methodology described in this paper supports participatory development of such outputs by employing design patterns as a communicational framework to support an inclusive and interdisciplinary community of learners, teachers, researchers and designers / producers of technology and/or content.

1.1 TEL as a design science

A thoroughly tested interdisciplinary methodology for TEL is a long-term goal. We believe a first step in this process is to build interdisciplinary approaches into the TEL design process. Design, in this context, includes both pedagogical and technical perspectives. The primary rationale for beginning with the design process is that it is of fundamental importance to the development of TEL tools and their associated learning activities. For example, understanding how TEL tools can be used in educational settings also entails familiarity with the pragmatic constraints of these settings. Design decisions in the subject dimension pertain to the question of selecting and connecting subject content – i.e. of designing supporting structures. The question of pedagogy is a question of designing instructional structures; the question of software engineering is a question of designing technological artefacts; and so on. We see all these aspects as various facets of *design knowledge*. While each party may have expertise in several of the associated knowledge domains, no single party has expertise in all of them: interdisciplinarity is required (Bannon, 1992).

For many researchers, design has become a ubiquitous activity. We trace this focus back to the seminal work of Simon (1969), who was the first to refer to design as a science. Simon distinguishes between the natural sciences and the sciences of the artificial, challenging the view of the latter as 'practical' science or 'vocational arts'. At the core of the study of the artificial, Simon places the science of design. In his words, "*everyone designs who devises courses of action aimed at changing existing situations into desired ones*" (Simon, 1969, p 129).

In Mor & Winters (2007), we identified three key elements in Simon's approach: a proactive, value driven scientific agenda; a functional axis of decomposition; and the relevance of representation. These three elements, while not always clearly acknowledged, are threaded through much of the work in TEL, and in particular in the design-based research tradition (Barab & Squire, 2004; DiSessa & Cobb, 2004; Brown, 1992).

In terms of scientific agenda, where natural science asks what *is*, design science asks what *ought to be*. When design science addresses social subjects – such as learning or development – the *value* aspects becomes visible. Neurobiology and psychology are concerned with how humans learn, whereas the science of Education asks how they *ought to learn*. The first may claim to be value neutral and objective, but the questions of education, by their imperative nature, are evidently derived from the observers' (often implicit) ethical, social and community agenda.

All the sciences proceed, to an extent, by decomposing complex problems into simpler ones. Design science is interested in purpose, intent and the shaping of the world to these ends. A functional axis of decomposition means analysing systems by what they do rather than how they are structured. This principle is salient in approaches such as activity-theoretic interaction design (Kaptelinin & Nardi, 2006) and learner-centered design (Soloway, Guzdial & Hay, 1994; Quintana et al, 2002).

While rarely in direct reference to Simon (with the notable exception of Kafai, 1995), many studies highlight the issue of representation and its importance for learning. This issue has been noted as key to paradigms such as constructionism (Noss & Hoyles, 1996), informant design (Scaife, Rogers, Aldrich, & Davies, 1997), and semiotic mediation (Radford, 2000). Balacheff and Kaput (1996) provided an extensive review of TEL tools for mathematics, and highlighted the continuous effort to diversify representations. Ainsworth et al. (2002) challenged common assumptions regarding the unconditional educational utility of multiple representations, arguing that it is strongly contingent on the design of the representing world as well as the represented one, and the relationship between them.

1.2 TEL as an interdisciplinary field of study

We believe that interdisciplinary is at the core of TEL research and practice. However, research in particular has traditionally been very discipline-oriented, exemplified by Universities structured primarily into Faculties and Departments. By contrast, today's society demands interdisciplinarity and application-oriented knowledge production from its workers. This point is being addressed at a policy level, where "funding agencies are increasingly stressing the social relevance of research results, and consequently a new mode of application-oriented research is emerging" (van den Besselaar and Heimeriks, 2001). Gibbons et al. (1994) draw on this distinction between disciplinary and non-disciplinary approaches, briefly summarized as follows:

- Mode 1 is focused on the production of traditional disciplinary science and discovering the laws of nature working within a well-bounded and specific paradigm. Examples include physics and chemistry;
- Mode 2 is focused on interdisciplinary and application-oriented knowledge production through the study of artifacts and systems. Examples include computer science and biotechnology.

Van den Besselaar and Heimeriks (2001) go on to define non-disciplinary research as "ways of combining elements from various disciplines, as an interaction among two or more different disciplinary specialties, in order to answer practical questions and to solve practical problems".

We class technology enhanced learning as non-disciplinary research in that it draws from multiple fields including the subject domain (e.g. mathematics), design, software engineering, teaching, psychology and computer science. The nature of the interaction between these disciplines, amongst others, within TEL is complex and varied. However, two main approaches can be identified: *multidisciplinary* and *interdisciplinary*. The main differences between them concern the levels of integration and communication. When taking a multidisciplinary approach, participants maintain their own disciplinary approach to the problem. Each development is undertaken by a 'discipline expert', effectively creating silos within the team that can lead to little or no integration. On the other hand, when taking an interdisciplinary approach, the aim is for coherence, where participants work together on activities in an interleaved, iterative and integrated manner.

1.3 Patterns as a support for interdisciplinary design

To support the interdisciplinary design process within TEL, mediating scaffolds are required. Primarily these act as ways for participants within a team to gain common ground through the communication of design knowledge. We argue that *design patterns* hold a powerful promise for recording, calibrating and collaboratively refining expert knowledge. Patterns are flexible enough to address a very broad spectrum of practices, from in-depth technical development to deployment issues in classrooms. In addition, they are rigid enough to oblige the pattern writer to focus on and concisely capture abstractions of their own best practice. The *pattern language* approach (Alexander, Silverstein & Ishikawa, 1977) was developed as a form of *design language* within architecture. This approach has been embraced in several other disciplines, including software engineering (Gamma et al., 1995), hypermedia (German & Cowan, 2000), interaction design (Erickson, 2000; Borchers, 2001), education (e-learning systems (Derntl & Motschnig-Pitrik, 2005), the design of computer science courses (Bergin, 2000) and computer games (Bjork & Holopainen, 2004).

Patterns support interdisciplinary practice in two main ways. First, from their inception they have been viewed as a mechanism for community participation in projects, making a process "so explicit that anyone can do it" (Alexander, 1979 p.10). Alexander promotes that idea that pattern languages have the explicit aim of externalizing knowledge to allow accumulation and generalization of solutions and to allow all members of a community or design group to participate in discussion relating to the design. More recently, Dearden et al (2002) proposed the 'facilitation' model (developed by Alexander et al (1985) in the Mexicali project) for participatory design. In that project, an 'Architect-builder' worked with a family to enable them to design and build their own house. Very significantly, the pattern language was shared by the designer and the family, and used to present and discuss design problems and solutions.

Furthermore, as we envision it, the *process* of pattern language development supports interdisciplinarity. Patterns are developed in an iterative, bottom-up manner, in which all perspectives are considered. Each pattern is rooted in the expertise of one or more participants, but then needs to be woven into the fabric of the language, thus representing the aggregated knowledge of the community. This results in patterns being refined and renegotiated, and new patterns and categories identified within the language. For us this reinforced the notion of a pattern as "something in the world" (Alexander, 1979 p.181), a dynamic entity which the community is free to iterate and amend in a participatory manner. In doing so, as the language grows it reflects the way in which its structure developed. Languages developed in a non-interdisciplinary manner will be biased in a particular direction: for example, within the context of TEL, they may be heavily weighted toward classroom deployment, ignoring technological issues. Our language (http://lp.noe-kaleidoscope.org) addressed this issue from the outset through the use of case studies and typologies which reflected the expertise of all members of the project team. More details on this aspect of our work can be found in Section 3.

1.4 Interdisciplinarity and the design cycle

McAndrew et at (2006) stress the knowledge-building and educational roles of patterns. They argue that the context component guards against over-generalization. The patterns should be written in a way that teaches novices how to identify the key issues in their domain and adapt the solution scheme to their specific circumstances. Drawing on Schön and Bennett (1996), they propose the notion of design as 'conversation with materials': "a process in which a designer makes a number of more or less tentative design commitments, reflecting on the emerging design/artefact and retracting, weakening or strengthening commitments from time to time" (McAndrew et al, 2006, p 221). Given the complexity and diversity of the issues to consider, the designer's focus must constantly shift from one part of the problem to another. Yet the inter-relatedness of these issues demands that they all be considered coherently. Design patterns can resolve this contradiction by providing high-level 'roadmaps' for design.

Our approach extends the above discussion in two dimensions. First, we question the expert-novice dichotomy, and move away from the 'knowledge delivery' perception of patterns. Second, we challenge the 'lone designer' paradigm in order to recognise the multi-voiced 'conversation with materials' that is more suited to the interdisciplinary nature of TEL.

Following the design research tradition, we observe a cycle in which theory informs design and use, and vice versa (see *Figure 1*). Each of the phases in this cycle is dominated by a different segment of the community, but ideally invites the participation of all. Design patterns support knowledge sharing in various transitions in this cycle. For example, David et al (2006) propose patterns for learning evaluation. Most pattern language efforts see the main function of design patterns in the transition from a field's theory to the practice of design. We also see patterns as a tool for theorising design knowledge from analysis, and as being applicable to the ways technology is deployed and used – not just produced.

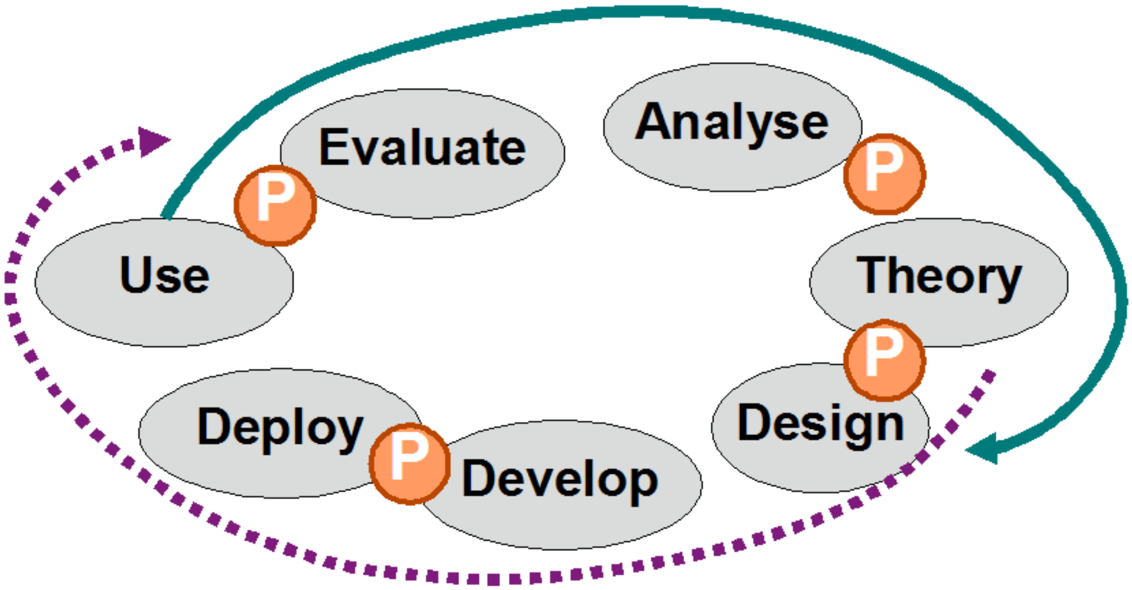

Figure 1: The design-research cycle. 'P's indicate the phases where patterns come into play.

Different participants enter this cycle at different points. A teacher might choose an instrument to try out, and design a method for using it in class. The teacher or a researcher might use known patterns for evaluating instrumental use. If the evaluation is favourable, the method would be adapted, in a participatory manner, to other circumstances and eventually analyzed to identify patterns. A researcher might take such patterns as data, correlate them with educational theory, and derive more elaborate patterns. These might be taken by a technology or learning designer as a starting point for producing new instruments. Ideally, this cycle functions as a relay race, where in the transitions between phases leadership is passed to the party with the greatest relevant expertise but all are involved. However, all participants are involved at all stages. From our experience, this requires intense design-level communication within the team, a role that a collaboratively constructed pattern language can support. An example pattern used in the 'theory-design' part of the cycle is presented in Section 2.5.

This design cycle provides a lens through which to view interdisciplinary practice. At each stage of the cycle, the emerging patterns capture various facets of design knowledge within the team. They evidence the nature of interdisciplinarity. Furthermore, patterns emerge in an iterative, participatory manner between each of the stages, indicated by the 'P's in *Figure 1*. This provides a multistage mechanism for critique and reflection. Pattern language development is a community venture.

1.5 Pattern language development as a community venture

In our interdisciplinary work on the 'learning patterns' project, the *construct* of a pattern became our central language (whether discussing the design of TEL tools or their deployment). Arguably then, an most important facet of a pattern language is its potential as a framework for discussing and collaboratively refining design. In fact, this is precisely why it is called a pattern language, and not collection or set. Yet the process of eliciting design knowledge from a community is far from trivial. As noted by Goodyear (2005):

> Forming a pattern language … involves painstaking, iterative work, travelling in two directions. From the bottom up, one can sketch individual design patterns, to capture recurrent problems and solutions from our collective experience …, interpreting these also through the lens of research based evidence and theory. From the top down, one can try to structure the problem space of design, scoping out the largest and smallest patterns, and sketching relationships between patterns (written and as yet unwritten). Neither approach is sufficient on its own and each can lead to contradictions and problems for the other - hence the need for iteration, revision, patience and a tolerance of ambiguity.

Retalis, Georgiakakis & Dimitriadis (2006) propose a four-step method for eliciting a pattern. They note that most of the related literature highlights the usefulness of patterns, or presents specific specimens, but neglects the process of collecting (mining) patterns. While we were not aware of their observations in the course of constructing our language, we encountered many of the same issues during our project. Retalis, Georgiakakis & Dimitriadis (2006) refer to Baggetun, Rusman & Poggi (2004) for a summary of inductive versus deductive pattern mining, and Kreimeier et al. (2002) for those interested in collecting game design patterns. However, we agree with Goodyear (2006) that the process of pattern elicitation must be inherently iterative. The language must be allowed to evolve as a cohesive ecology of ideas. Whenever a new pattern is introduced, it perturbs the structure of the language and modulates existing patterns. For us, as distinct from existing 'out there' to be mined, patterns are socially constructed.

These issues are not limited to pattern languages. Kali (personal communication) reports similar challenges in compiling the design principles database (Kali, 2006). In particular, she notes the difficulty of inducing experts to contribute their knowledge. This is not just a matter of time (and the better the expert, the busier she is). The most valuable patterns an expert possess are (almost by definition) so entrenched in her practice that they have become second nature, and she is unaware of their significance. This corresponds to the well-known problem of attempting to elicit tacit knowledge (Friedrich & Van Der Poll, 2007).

2 The IDR methodology for interdisciplinary design

Popper wrote that "We are not students of some subject matter, but students of problems. And problems may cut right across the borders of any subject matter or discipline" (Popper, 1963). It is within this spirit that we have developed a *participatory methodology for interdisciplinary TEL design*, termed IDR. IDR is simultaneously a tool for and a product of design research. It offers a framework for collecting, communicating and iteratively enhancing design knowledge. This framework itself is refined by iterative small-scale, highly reflexive experiments, and validated by ethnographic, socio-cultural methods of action research. Moreover, IDR is designed to work within the design cycle outlined in Section 1.4.

In support of interdisciplinary practice within TEL, our aim is twofold: i) to engage participants in the process of reflecting upon their previous practices (Schön, 1983) which resulted in successful outcomes when used; ii) to scaffold this reflection as a process of abstraction to generalizable solutions useful to the wider community. The first aim allows participants in the team to share specialist knowledge working towards collaborative understandings. The second addresses the need to develop solutions with input from many disciplines, so as to be of broad use to the TEL community. To do this our overarching process is the development of patterns, divided into a three-stage *identification – development – refinement* (IDR) methodology. The first stage is to identify potential patterns through the use of typologies and case studies. The next stage is to develop a set of patterns based on designs evidenced by the case studies. Once this initial set has been chosen, the third stage is improving upon the patterns through collaborative discussion and reworking using our suite of web tools. The key point is that patterns *mediate* the interdisciplinary design process through their identification, development and refinement by the project participants. This methodology was used and developed within the 'learning patterns' project and concurrently iterated at five international conference workshops.

2.1 Stage 1: pattern identification

Before project participants can begin to work in an interdisciplinary manner they need to conceptualise their own area of expertise in a way that is accessible to others. To aid this process we developed *typologies*, visual mindmap overviews of particular domains. The typologies act as a semantic starting point for discussion around interdisciplinary design and capture particular aspects of TEL design knowledge. Each typology can be seen as a hierarchically structured glossary of one dimension of design knowledge pertaining to the problem domain at hand. The choice of dimensions would vary across domains. In our case, we focused on games for mathematical learning, thus the dimensions we choose where: *mathematical content*, *learning and instruction*, *educational context*, *games*, *interaction design* and *software design*. A different community, for example one focused on interactive books for history, would probably choose a different content dimension but might share the learning and instruction dimension.

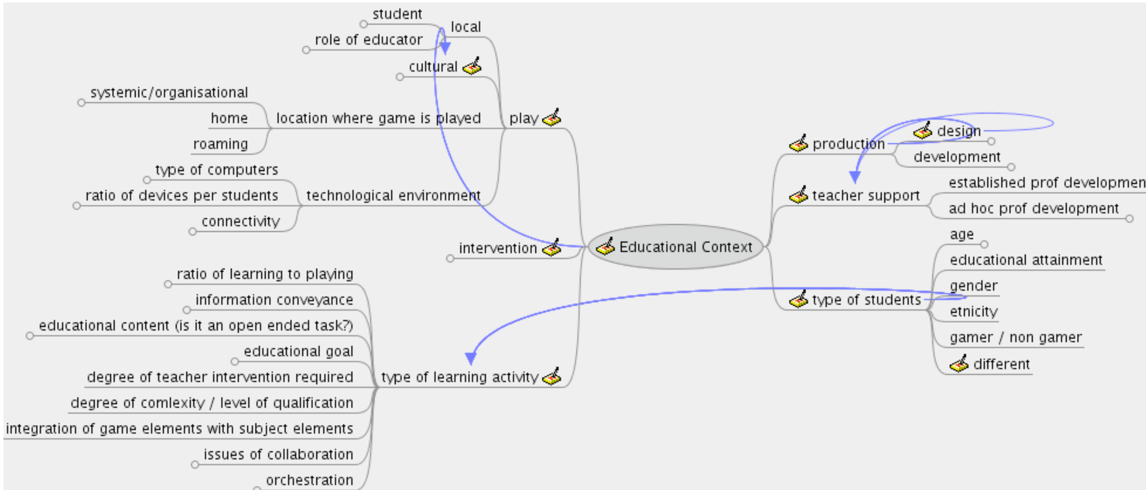

Figure 2: An example of one of our six typologies

The typologies were developed by a group of domain experts through an iterative process of construction, testing, negotiation and refinement. We initiated this process through a brainstorming session conducted during a project meeting. This provided the initial outline and candidate typologies. These were whittled down and following this session, domain experts published an initial draft of their respective typology online using an earlier version of the typology tool. These drafts were scrutinised by the other project members. Using an online discussion mechanism, we queried each other for clarifications and illuminated possible gaps and overlaps. This led to the typologies being iterated through a synergistic collaboration between all partners.

The next step was to explore the potential capacity and use of the typologies, by using them in the process of drafting small *case studies*. These were descriptions of real world practice, of which we have 24 in all. The rationale for this approach was our belief that for the typologies to be a productive tool, they needed to be refined through productive use. Each of the participants presented a case study. The purpose of the case study development is multifold: i) to provide concrete examples of practice within disciplines; ii) to map practices and content detailed in the case study to the set typologies; iii) for the team to identify linking points between disciplines; and iv) to provide the starting point for pattern development.

Case studies resulting from partners real world experiences (in both development and deployment) are presented and discussed. The need to accommodate the concerns of diverse design partners drives the author of a case study to identify the critical elements in their TEL design process, with respect to what design decisions worked and why, reflecting key choices that were made. In the process the discussion is gradually shifted from a specialized perspective to an interdisciplinary one. For example, a case study presented by a software designer might focus on software design, but discussing it with a teacher would bring out issues of activity design and fit to educational context. A fundamental aspect is to discuss not only positive steps and choices in the presented case, but also mistakes that may have been made and the rationale for why this was. The typologies are used as an analytic tool to frame the case study with respect to the various dimensions. At the same time, the discussion illuminates gaps in the typologies and drives their refinement. At the end of this stage, the critical design decisions identified by the participants are the initial set of potential patterns, reflecting expert thinking.

2.2 Stage 2: pattern development

From this set of design decisions, the process of developing or 'distilling' a pattern begins by ranking the decisions in order to identify a single design element which contributed to the success of this case. This element is then phrased in a manner which detaches it from the single example, but avoids over-abstraction. In the words of one of the project members, it is a "situated abstraction done by an expert". The pattern is carefully named: names need to be descriptive, concise and attractive. Its details are then moulded into the pattern template (see Section 2.2.1). This process is then applied to the remaining design decisions, as necessary, resulting in a first draft of a pattern set. Once each of the patterns in the set has been described, they are mapped to other case studies and the comparison used to refine the pattern's critical features. This may lead to the need to define new patterns - as special cases of this one or as generalizations of it. For doing so we define four types of relationships: *Elaborates*, *Elaborated by*, *Follows* and *Leads to*. Elaboration defines an 'is a type of' relationship, similar to class inheritance in object oriented programming. The 'elaborates' / 'elaborated' links define a single-root, multiple inheritance hierarchy of abstraction. Patterns are also grouped into categories, defining a coarser-grain tree hierarchy. 'Follows' and 'leads to' define lateral connections. These could be thematic (pattern B is useful in the same context as A), temporal (after using pattern A, consider using B) or structural (super-pattern C is composed of patterns A and B).

2.3 Pattern Structure

One of the key objectives of developing a pattern language is the standardization of design knowledge. As a result, the task of defining a pattern template is common to many pattern language initiatives. Alexander defines a pattern as a "three part rule, which expresses a relation between a certain context, a problem, and a solution" (Alexander, 1979, p 247). Each community elaborates this structure in a slightly different form. Our template includes the following sections:

- Name: 3-4 words, catchy and descriptive.
- Metadata (author, entry date, last edit date, category, and status)
- Short summary: one or two lines.
- The problem / intent: what is the problem or need that this pattern addresses.
- Context: where is it applicable?
- The pattern: 'cookbook' description of a possible method of addressing the specified problem in the specified context.
- Related patterns: vertical and lateral links, as described above.
- Examples: links to concrete case studies where this pattern is manifested.

- References: note of academic or other resources pertaining to this pattern.

Choosing a good name is important because it makes the core idea of the pattern explicit in a compact and easy-to-remember manner for users. This name is the pattern's identifier in verbal and online design discussions, as well as the key for searching or browsing the pattern language site.

The short summary allows visitors to quickly evaluate their interest in a pattern. It also plays an important role in the pattern refinement dynamics described in the next section: when a pattern emerges in the course of discussion, its initial record it will include only its name and summary. Further details are added during Stage 3.

The context includes a narrative of the circumstances in which the pattern was identified, and a mapping to the various typologies. In the case of simple patterns, the solution body itself may take the form of a sequence of numbered steps for users to follow to operationalise it. For more complex patterns, diagrams and detailed descriptions will be added as well as implementation notes, links to sub-patterns etc.

The relationships between patterns are listed as described above. This is important as it defines the networked structure of the pattern language. Thus, by the end of this stage, although some very specific pattern template details may need to be completed, we have moved from a loosely defined patterns set, to a hierarchically structured and mapped pattern language. The pattern language is organized hierarchically. The top layers of this hierarchy are abstract categories of patterns, while the lower ranks are concrete patterns and sub-patterns.

2.4 Stage 3: pattern refinement

We view our language of design patterns as a dynamically evolving resource, and this vision is reflected in the structure of the language and in the tools (see Section 3), which support it. At this stage, all patterns are classified at having one of four states: seed, alpha, beta and release. Seed patterns often represent design ideas, decision and practices, which were noted during collaborative discussion in the identification stage (or while developing other patterns in the development stage). Essentially they are placeholders, ready for development. This does not undermine their importance as they capture potential patterns, directly addressing the "cold-start" problem of identifying patterns (Retalis, Georgiakakis & Dimitriadis, 2006).

A seed pattern is often meaningful only to its immediate authors. Following its identification, the authors need to elaborate it to a level of detail such that the other members of the community can understand it. This would transform it to an alpha state, which signifies patterns that are undergoing continual internal discussion by the project community. However, they still require some refinement before they are submitted to public review. This refinement would include completing missing details, elaborating the context, checking and refining the links with other patterns, and adding any graphics or formatting which will make the pattern easier to read. Refinement is a focused process, requiring a detailed review of the pattern with a particular emphasis on the quality of the problem statement and the generalizability of the proposed solution. These are absolutely critical to a pattern's success and so should have a clear relationship. As distinct from the 1-to-1 'shepherding' relationship suggested by Harrison (1999), our refinement process is community based, involving all members of the interdisciplinary team.

Once this work is completed, patterns are marked as beta, which means they are open for public review on the website. Many patterns in this state will face little significant further refinement. However, public review is very useful for gaining an insight into the clarity of the patterns, how well they communicate their intent and how understandable the links between patterns are. Furthermore, public review can also catch some discrepancies in language. The feedback from this review is used to bring the patterns to their final release state, i.e. as the output of Stage 3 of the IDR methodology.

2.5 An example pattern

Our pattern language consists of over 120 patterns, available for use under the Creative Commons licence (http://lp.noe-kaleidoscope.org). While it is not possible within the context of this paper to provide details of the complex interrelationships between the patterns, we evidence our work here by providing an example: *First Boundary Prototype*. This pattern addresses the question of how to determine the boundaries of a first software prototype and involves input from educational researchers, interaction designers and software developers. The aim is to develop a prototype that can act as a mediating artefact for discussion when teachers and learners are brought into an iterative design process at the next stage. As noted in Section 1.4, the pattern works at the interface between theory and actual design.

In summary, this pattern addresses the problem of interdisciplinarity in three main ways. First, it works at the interface of the fields (game development, interface design and education) involved, thus requiring a) input from different members of the team and b) understanding of perspectives outside of your own domain. Second, it solves a problem common to all three, namely development of the first prototype. This provides a clear rationale for its use. Third, it provides new insights into practice by taking into account the perspectives of all three participants, enabling participants to work towards a common ground.

Name: First prototype boundary

Problem: *How to determine the boundaries of the first prototype collaboratively*?
This pattern address the problem of how to delineate the scope and depth of the first game prototype developed. It is motivated by the need to have a working version to mediate discussion around the next iteration required.

Context: The pattern emerged from a context where educational researchers, interaction designers and software developers are building the first prototype. The objective is for participants to communicate their needs and expectations at this stage. If communication between parties is not clear, prototype boundaries cannot be set productively.

- Subject content: skill domain.
- Learning and instruction: modalities of employment, approaches and theories.
- Educational context: play, intervention, type of learning activity.
- Games: game as activity, game as social function.
- Interface and interaction: user interface.
- Software design: development methodology, platform.

Pattern:

1. The educational researcher bounds the design by providing an initial sketch on paper. This delineates a simple initial exemplar scenario for development. It provides an interaction metaphor to be discussed and shows how the tool design is underpinned by learning theories, educational context and subject content.
2. The researcher, designer and developer discuss and agree on the simple exemplar tool to be implemented. This will encapsulate a single aspect of the tool that will involve a sequence of interactions common to the full version.
3. The designer proposes what the interaction metaphor means in practice and how the interface can be designed to address it. Particular emphasis is placed on component interaction, display modes and potential usability issues.
4. The game developer details a first version of the system architecture and appropriate programming language for the tool to be implemented in. Details of how the front and back ends of the tool will communicate are discussed and agreed upon.
5. The final output is a bulleted design document for the first prototype with a schedule for completion.

Therefore, wherever possible the team works towards communicating the initial boundaries from the perspective of each community and thus develops a joint understanding about what that means for constraining the first prototype.

Related patterns:
Follows: Knowledge driven design.
Elaborates: Team communication, Iterative design, Experimental design, Interaction design, Educational context.
Elaborated by: Learning activity, Programming language, Menu design, Window Transition.
Leads to: Interactive components.

Category: Design

Table 1: The First prototype boundary *learning pattern addresses the need for effective communication to constrain the first prototype.*

3 The IDR tools

Alongside the development of the pattern language, we have developed a set of interactive web-based tools to support it (see Mor et al., 2007 and http://lp.noe-kaleidoscope.org/outcomes/patterns/map). The primary functions of these tools are to allow us to manage the pattern language efficiently, and at the same time make it easy to use by any interested reader. These tools provide various methods of browsing, reading, editing and organizing typologies, case studies and patterns. At a more significant level, these tools supported our practices and allowed us to experiment with, elaborate and extend the IDR methodology.

Rabardel (2003) and others (Artigue, 2003; Guin & Trouche, 1999) describe the dynamics of *instrumental genesis*, by which learners first master a tool to make it an instrument (*instrumentation*) and then adapt the tool to their needs (*instrumentalization*). While the theory acknowledges the socio-cultural context, it focuses on the relationship between the individual and the tools offered by this context. Yet when considering technologies which relate both to the individual and the social aspects of learning, it is useful to draw a parallel and consider dynamics of *social instrumentation*; observing the processes by which a community masters new tools, appropriates them as collaborative instruments, and then modifies them to meet its needs. Such tools can be artefacts (physical or digital) as well as practices and even norms. The co-evolution of our suite of web tools and the IDR methodology is an example of such dynamics in action, exemplifying iteration around the design cycle (see Section 1.4).

These dynamics were accentuated by the need to find a common language in a diverse community. The patterns, the methodology and the tools became boundary objects (Star & Greisemer, 1989; Star, 1990) connecting the various elements in a combined interdisciplinary effort.

A structural specification of these tools is available in the project report (http://lp.noe-kaleidoscope.org/). In this paper we only note some of the main features, with emphasis on their role in supporting the implementation of the IDR methodology. Hence these features are listed not by component but by function, in order of use.

3.1 The typologies tool

The typology tool allows community members to browse, review and edit the various typologies. A typology is a complex monolith which captures the knowledge of a single specialist (or specialist group). Hence it is convenient to edit it off-line and upload versions as they mature. This is achieved by using a mind map editor (FreeMind). Once a typology is uploaded, it can be viewed either as a map image or as an html tree. The definitions of the terms are displayed alongside the tree view, and can be edited online. Each typology map is accompanied by a discussion forum, where other members of the community can comment, suggest changes or ask for clarifications. When a new version is uploaded, the previous versions are retained for reference. The most recent version is also displayed in the 'outcomes' section of the site.

3.2 The case study repository

Case studies are created, edited and indexed online using a simple template and a visual editor. Contributors create a new case study by providing a name and a short summary. They are then directed to an editable online document based on the case study template. This template prompts them to provide the context, aims, details, outcomes and references. The main bulk of the case study is expected to reside in the details section, which is a free-form narrative. Contributors are encouraged to include graphical materials, such as screen shots and diagrams. Our templates are all 'soft templates': they offer a structure, but do not impose it. The contributor has full artistic licence to describe her case study in any way she finds most fitting.

To facilitate ease of referencing to typologies, the editor supports wiki-style 'quick-links': entering the text [[T6:Requirements]] will create a link to the requirements node in the software development typology.

3.3 The pattern editor

The transition from the identification to the development stage of the methodology is marked by populating the pattern repository with seed patterns, derived from the case studies. This is achieved by providing a name, short summary and category. Once this is done, the user is directed to the webpage generated for the new pattern. The meta-data for this pattern is automatically generated and listed in the page header. Below are empty sections for the different pattern elements, as described in Section 2.3. As with the case studies, this is a soft template: pattern authors may override them, although they would rarely do so. As with the typologies and case studies, each pattern is accompanied by a discussion forum to facilitate its refinement by the community. Pattern editing and publication is a two-phase process: an author can edit and save a draft of the pattern without affecting the publicly viewable version. Once she hits the 'publish' button, the current version is archived and the draft is published.

3.4 The pattern browser

Once a pattern has been added to the collection, it can be found, edited, and woven into the fabric of the language through the pattern browser. The pattern browser is the central tool in our system. It provides several modes for viewing the patterns, as well as entry points to tools for creating new patterns and structuring the language. It is critical to stage 3 (refinement) of the methodology.

When a new pattern is entered into the database, it is automatically listed in the table view, which can be sorted by various keys. As part of the refinement stage, the pattern author needs to map its relations to other patterns. She does so by first assigning it to a category through the pattern editor.

Looking beyond the single pattern, the structure of the language as a whole can be edited as a FreeMind mindmap. Community members reviewing the language start from the overview mode, which displays the main branches of the pattern hierarchy as an image. They then switch to browse mode and use a tree-based navigation tool to hone in on the patterns they wish to discuss. The structure of the language as a whole can also be discussed in a forum adjacent to the browser. Alternatively, reviewers can use live mode that utilises a java applet to browse the map, combining the functionality of both overview and browse modes.

To eliminate the risk of 'orphaned' patterns which are not linked into the language structure, the Index (table) mode lists all patterns in the database, even those in a preliminary state which have not yet been integrated into the structure. Index mode displays each pattern's name, author, creation and last modification date, summary description, category, status and ranking. The table can also be sorted by each of these columns.

3.5 Epilogue: trails

As more expert knowledge is embedded in a pattern language, it becomes more intricate. Consequently, there is a significant likelihood that the patterns, and critically the relationships between them, become less accessible to novices. In the worst case, some members of the community that created the language find it increasing difficult to use it. Novices do not know where to start and how to penetrate it, as the structure of the language does not expose the path along which it evolved. In an attempt to address this issue, we have developed a tool called '*Trails*' (http://lp.noe-kaleidoscope.org/outcomes/patterns/trails). A trail is an informal illustrative account of how patterns were derived or how they might be used. The purpose is to provide a starting point for detailing a particular practice that the pattern language covers (for example “beginning the design process”) in narrative form, providing links to each of the patterns used. The aim is not to present the narrative as hard data or detailed analysis, but rather as an aid for the reader to gauge the nature of the patterns approach. It offers an initial opportunity for readers to begin to understand the deep, complex and structured relationships between patterns, while knowing that these relationships can, and have been successfully explored and mapped in an interdisciplinary manner. Furthermore, trails allow for exploration at both the abstract and specific levels by constructing the narrative to ‘drill-down’ through the levels of the language hierarchy.

4 Results: developing and using the IDR methodology

There are three main results from the implementation of the IDR methodology within the ‘learning patterns’ project:

- The pattern language and the development tools
- The workshop model
- The importance of the educational context to pattern development

4.1 The pattern language

The heart of the IDR methodology is interdisciplinary pattern development. Our pattern language was developed by a core project team of 21 participants, distributed across six European countries, further supported by three partner schools. Expertise within the team included software development, gaming, pedagogical practice and interaction design. Furthermore, the patterns were iterated and validated during five workshops at international conferences, which were attended by over 200 delegates from the academic, industrial and practitioner communities, all of which received highly positive reviews (see Section 4.2).

Over the course of the project, participants shared design knowledge by abstracting and generalizing from their real-world experiences. The first indication of the viability of our methodology is in the volume of content and activity it has generated. In the course of a year, we have developed over 120 patterns (see Table 2 for a listing), 49 of which have matured to *Beta* or *Release* state. The patterns deal with both design and deployment issues within TEL and in particular many focus on the interface between them, exemplifying the interdisciplinary nature of our work. Moreover, the language itself displays a complex networked structure, the result of close to forty iterations of refinement. Most patterns in alpha state and above have five to ten recorded versions, with some displaying up to twenty (of course, the actual number of edits is much larger). Our case study repository, the starting point for pattern development, includes twenty-four instances almost half of which were contributed by workshop participants.

* -- Patterns
o -- Methodology
+ -- Iterative design
# Event-driven iterative design
# Layered Research-Design Iterative Cycles
# Low Fidelity Prototyping
+ Participatory design
+ -- Experimental design
# Tandem
+ -- Related Knowledge Collections
# -- Software Design Patterns
* + GoF patterns
o + Creational Patterns
o + Structural Patterns
o + Behavioural Patterns
* Stream
* Model-view-controller
o -- Design process
+ -- Bootstrap
# -- Knowledge-driven design
* First boundary prototype
# -- Metamorphosis
* Content morph
* Rejigging
# Design Exploration through Gameplay Design Patterns
# Extreme Characters
# Tarot-based Design Inspiration
+ Remap
+ Play-test-use-eval
+ Personas
+ Wrapper-intended Design
+ -- Concept Development
# -- Storyboarding
* Observation
# -- Content Embedding
* Content Embedding in Rules
* Content embedding in toy
+ Layered design

o -- Structure
+ -- Construction
# -- Microworld
* Microworld design
* Game with microworld
* Half - baked microworld
* -- Undercover process
o Fixing the Tool
Mathematical game-pieces
Task in a box
Designer bug
Hard but not too hard
-- Production
* Problem posing
# Purpose and utility
+ -- Communication
# Something to talk about
# Objects to talk with
# Build-talk
# Performance
# Semi-automated meta-data
# Encouraged Retrial
# Post ludus
# -- Soft scaffolding
* Active worksheet
+ -- Collaboration
# -- Challenge exchange
* Guess my X
# -- Programming as game
* Build this
* Use this
+ -- Narrative
# Narrative spaces
# Narrative representations
+ -- Orchestration
# Drill & Argue
# Crescendo
# Chain of mini games
# -- Sustaining interaction
* -- Visualised social dynamics
o League chart
o Axes of participation
* Audience awareness
+ -- Epistemic software patterns
# EP-Stream
# EP-MVC
# EP-Facade

Table 2: An overview list of the patterns in our language

Although the project has officially ended, our patterns website (http://lp.noe-kaleidoscope.org) still receives over a thousand unique visitors every month. As a result our language is constantly evolving, a testament to both the interest of the TEL community in the issue of participatory design but also to the use of our web-based pattern development tools.

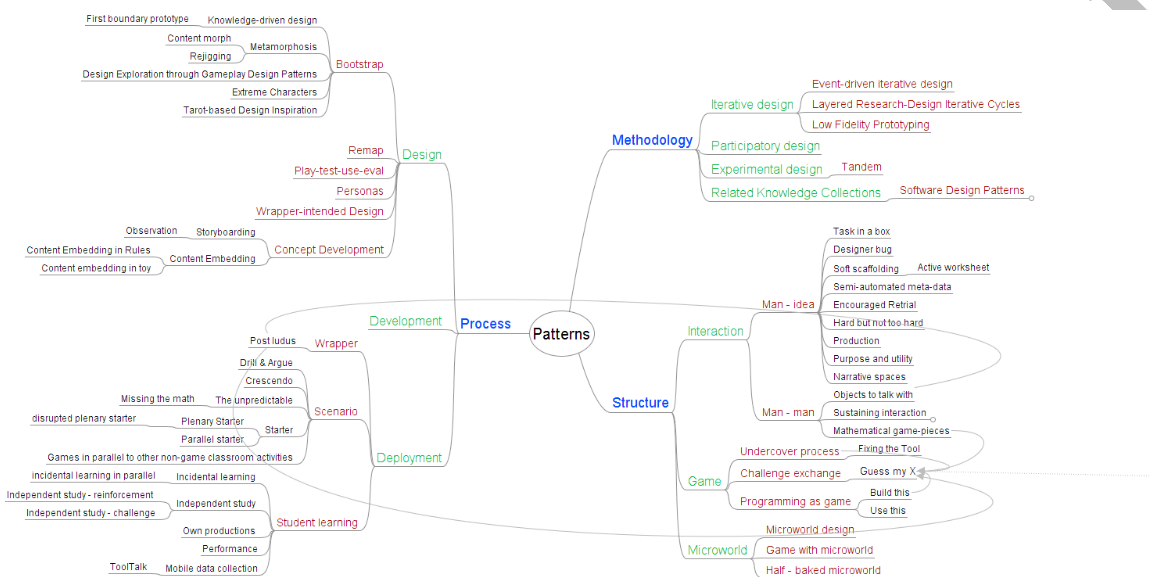

Figure 3: Overview map of our pattern language

The web-based tools (http://lp.noe-kaleidoscope.org/outcomes/patterns/map/) we developed allowed us to collaborate across geographic, cultural and disciplinary barriers. As noted above, the project team was distributed across six countries, and workshop participants came from even further afield. Apart from the several days of project meetings and workshops, all pattern development was facilitated by online collaboration. This would have not been possible, in such a short time, unless we had taken the decision to adopt an iterative approach, allowing the tools and the methodology to co-evolve. In particular, the different views of the pattern language, provided by our pattern browser, allowed us to maintain control over a complex, ever-changing body of knowledge. The distinctive support for authoring, editing and discussing patterns allowed us to foster collaboration while maintaining authorship and coherence.

4.2 The workshop model

The workshop model was designed to engage a broad community in the collaborative development of seed design patterns (Winters & Mor, 2007). The model is briefly summarized as follows. Participants were contacted prior to the workshop date, and encouraged to contribute case studies from their own experience. On the day of the event, the primary focus was on the pragmatics of stage 1 and the initial steps of stage 2 of the IDR methodology. This decision was taken in order to ground the pattern approach in participants' everyday experience. We began by walking participants through the process, presenting exemplar case studies from our own research and detailing how we mapped these to our typologies using our web tools. We then facilitated small group work on the same activity, motivating participants' discussion of their own practices, reflecting on the commonalities and differences of their contexts. Once this stage was completed, participants fed back to the whole group and the facilitators noted generalizable design decisions in collaborative discussion with the group. The participants were encourage to critique and motivate why i) each design decision was chosen and ii) the process of how it could become a generalizable solution, resulting in a set of distilled seed patterns. Next, each group discussed the seed patterns, presenting how they might be used in their own contexts, referring, where appropriate, to the contributed case studies.

Reflecting on these workshops, we found that the tools and the ways we used them in our project framed the development of the workshop model for introducing others to the IDR methodology. Pre-event exposure to the tools, along with the use of social technologies (blogs and mailing lists) created a sense of familiarity and cohesion which allowed us to facilitate interdisciplinary discussions in the relatively short time at hand (ranging from a couple of hours to a day). The use of participant contributions ensured that the content of the workshops related directly to their interests and professional context. Indeed, we witnessed very lively discussions in the group work phase. On the other hand, the need to present your case study to a diverse audience prompted reflection in unexpected directions. These reflections where manifested in a burst of seed patterns emerging from each workshop – up to 26 in one case. The number of patterns distilled was clearly proportional to the workshop length. Unfortunately, many of these were not refined to a Beta or Release state. Based on the scaffolded nature of our workshops, we speculate that this could be due to lack of structure and resources to support *on-going* and *wider* community dynamics within participants' real-world contexts.

After each workshop, the methodology was updated to reflect the comments and difficulties of the participants. The major aspects that participants found valuable in i) aiding their understanding of the patterns approach, and ii) in beginning to develop seed patterns are as follows:

- An increased focus on practice: at the beginning stages we detailed in-depth the rationale and theoretical foundations of the design patterns approach. While this was welcomed by some participants, the majority of participants from both the industrial and educational sectors, were more interested in how the approach was able to help them pragmatically. This might have been expected as patterns are developed based on empirical evidence of what has worked in the past. This concern was common across the disciplines from which participants came and in later workshops proved a good 'bonding point' from which to work together.
- The 'hands-on' element of the workshop was highly valued. Giving participants access to the collaborative web tools in advance of the day allowed them to bootstrap discussion around their particular cases, providing a sense of relevance to their own work. Working in small groups and feeding back allowed for the comparison of different approaches while seeking potential commonalities which could be generalized.

- The workshops attracted participants from a wide variety of backgrounds. However, working in groups with people from different disciplines proved difficult for some participants. There was a need for an evolving negotiation between participants in order to (i) understand their respective positions, (ii) what was important in their everyday work lives and (iii) what were the pragmatic constraints they were working under. As facilitators, we needed to be aware that time was needed for participants to simply engage with each other, seeding fruitful and respectful discussions.

Preliminary indications from the feedback received from workshop participants were very positive. For the most part, they were happy to meet and engage with people from “research, design and educational backgrounds all together”. Others expressed concern that the workshop length (in this case, half a day) was too short but expressed the view that they would continue exploring the ideas in their work, through the use of the online suite of tools. However, this seemed to mostly happen only for those who were already experienced with the patterns approach, indicating a need to continue to work on ways of scaffolding novice practice. Many participants reflected on their own experiences and commented on the potential of patterns: “While I am familiar with the origins of the approach … it was interesting for me to see it applied so fully in a different field. The workshop will certainly make me re-think about the use of patterns in some of my work”. Some did mention going back from the workshop and attempting to use patterns with their teams but how this went was not reported. In any case this would require further, longitudinal analysis to probe fully. Reflections on our own development of patterns within the project team are provided in Section 5.1.

Many workshop participants saw design patterns as a practical tool, emphasizing their pragmatic, real-world situations. It was interesting that the importance of a holistic approach was evident to them: “I think a *collection of patterns* (emphasis added) could be a very useful practical tool in game development, … , potentially as a 'vocabulary' for user-centred design with children” and “I'd be interested to see how the collection of patterns evolves and to see it in a final edited form”.

Figure 4: Participants discussing pattern development at the workshops

4.3 The importance of the educational context

A key motivator for the development of design patterns is to improve the process by which technological innovations can serve educational needs. Therefore, the educational context within which TEL tools are deployed is critically important. Designers need to understand this context, not just in relation to the immediate settings but in terms of the broader social and cultural elements within which learners act and which influence their activity. This is precisely why the empirical element of deriving design patterns is so important – it gives voice to the critical design issues of importance at the 'coal-face' of the interdisciplinary team's practices. In this way, contextual issues, which might otherwise go unnoticed, can be engaged with early in the process.

Taking account of the context in TEL design requires both knowledge integration and communication between the participants in the design (Kynigos, 2002), which as discussed earlier, is what design patterns support. The context of design is explicitly embedded in the design patterns themselves – every pattern details the context in which it was developed and all patterns are related to one another. Therefore, in our experience of collaborative pattern development, the nature of the context was a fruitful point of discussion. All participants needed to come to an understanding of what context meant, not only in terms of the setting but also by taking into account the expertise available (for example, the technical expertise of teachers) and how any intervention challenged pedagogical practices.

Developing the typologies was a community inquiry into context. Our process of collaborative learning began from a need to negotiate the dimensions of context we focused on. The process continued through collaboratively elaborating upon the vocabulary and structure of the typologies and validating them through use in the development of case studies and design patterns.

In several cases we identified similar concepts addressed from different perspectives. For example, the social setting (individual vs. group work, etc.) has ramifications in terms of educational approach, classroom pragmatics and technical setup. Hence it emerged in the three different typologies. Such examples highlight the importance of the interplay between content design, activity design and tool design. Our methodology raised participants' awareness of the complex context of their case studies and patterns by requiring them to classify these along the axis of the different typologies. This requirement was promoted by building a structure of 'mapping to typologies' into the templates and editors. We identified a need to enhance this feature further by adding tools for quick browsing and selection of typology terms, which has yet to be implemented.

Over time, the pattern development process displayed a 'saw-tooth' dynamic, originating in practice, climbing up the hierarchy towards abstract theory, and drilling back down to implications of the theory for other branches of the language. This created a continuum between theory and practice which benefited both practitioners and researchers. For practitioners, starting from empirical cases provided a manageable entry point and a path into theory which would otherwise have been inaccessible. Researchers, on the other hand, were required to ground their theory – while avoiding the known pitfalls of this approach (Wastell, 2001; Bryant, 2002), such as the unproblematic conceptualisation of data.

5 Discussion

In Section 1.4, the design research cycle was discussed in relation to the ways in which participatory design can occur in an interdisciplinary manner. This illuminates a key consideration within interdisciplinary practice, namely that it is seen as important but can be difficult to practice. Outside of the oft-cited problem of intuitional barriers (Caruso & Rhoten, 2001), we found that the process of exchanging design knowledge is a difficult one. Here, we focus on three main points for discussion: the potential of design patterns for addressing this problem, the ways in which the social configuration of TEL development itself can help and finally reflections on what can be done to achieve interdisciplinary design.

5.1 Why design patterns?

Our study originated from a shared acknowledgment of the complexity and diversity of design knowledge which bears upon the practice of TEL development and deployment. If anything, this acknowledgment has been fortified. We strongly take the view of Goodyear (2005), in that pattern development "involves painstaking, iterative work". The 'saw-tooth' dynamic which emerged in our work recalls his description of iterating between bottom-up and top-down elaboration. We argued that a powerful way to sustain such a dynamic is through fostering true interdisciplinary community discourse through the IDR methodology. However, in developing IDR, we were forced to readdress the issue of why use a design patterns approach. Might, for example, simple one-to-one advice between those involved be enough? We concluded that the design patterns approach offers multiple advantages. However, here we will focus on the advantages in supporting interdisciplinary design through pattern *development*, as distinct from pattern use. The first observation is that in developing patterns, and in particular when developing the pattern map, participants need to begin to think in an abstract manner. When positioned as a basis for communication of design knowledge across disciplines, patterns force people to look beyond the specificity of their particular design process and focus on what can be distilled that is useful to the wider community. We make no claims as to the ease of this process – in fact most participants, in particular those from a non-technical background, found this process difficult. In a sense this validates the importance of context – for example, during the project teachers often stated that their particular context (classroom, age of students, student pairings) was critical to the success of any particular intervention they were making and therefore might not be generalisable. However, in the development of their understanding of patterns over time, they were able to better engage in abstraction of their processes. The specific details of a unique setting were replaced with a structural characterisation of the critical features of context.

Reflecting on our own development of a language of patterns, creating and understanding the nature of the pattern map demanded an appreciation by practitioners of its structural nature and by designers of its connection to learning activity and that the construction of such appreciations is non-trivial. To take into account the need for the language to be seen as a coherent whole involves understanding the nature of the relationships between the patterns, i.e. *Elaborates*, *Elaborated by*, *Follows* and *Leads to*. For those from a non-Computer Science background, which was most of the team, the idea of inheritances and multiple inheritances proved difficult. There was no way around this. As a team, we had to help each other work through the difficulties around this abstract notion in order to make the language fully accessible to the team. We did this by identifying patterns at higher levels of abstraction and clustering related patterns together in 'accessibility pathways', to reduce the number of possible entry points to the map of patterns.

A critique of any patterns approach is that it is inherently backward-looking: they can help you implement previous best practices but if what you are doing is very new, while they may help you a little, they are not ground-breaking. In the case of technology enhanced learning, we would argue that the vast majority of development and deployment is multidisciplinary in nature. There is a long way to go before the field can claim to be working in a truly interdisciplinary manner. Therefore, methodologies that help in achieving this must be seen as helpful to all participants involved. IDR is particularly strong in this regard as through pattern development, participants are required to understand and critique patterns outside of their discipline.

5.2 The social configuration of TEL development

Within the interdisciplinary research community, one of the primary impediments cited for the lack of real-world interdisciplinary practice is the social configuration of Universities around Departments. At a higher level of granularity, the configuration of TEL teams in general, is striking. While configurations do not necessarily focus on interdisciplinary practice, the situation might be said to be better than expected. DiSessa et al. (2004) conducted a comparative study of how research teams design, develop and evaluate TEL software, in the context of component-based educational programming. They identify the issue of the social configuration of the production team as "a critical family of issues that are easily marginalized" (p.117). DiSessa et al. (2004) study four such configurations in detail and note their relationship with the evolution of the technology and its use. These models reflect the different ways the various participants involved in TEL development are brought together. Based on the definitions of interdisciplinarity (van den Besselaar and Heimeriks, 2001; Gibbons, 1994; Committee on Science, Engineering and Public Policy, 2004), our methodology can be viewed as working most closely with the Integrated Team Model (ITM): participants from different disciplines work together in relatively small, product-oriented design groups. This method of working is reflected by the patterns in our language which focused on the interface between disciplines and included pragmatic ways to have teachers and technologists productively engage with each other. Furthermore, many patterns were developed from the use of particular tools in educational contexts, where the tools were developed from scratch as outputs of research projects. There was a reflection in the patterns of the need for participants to understand each others' practices in order to achieve integrated development. DiSessa et al. (ibid) reflect on the fact that with the ITM,

NOTICE: this is the author's version of a work that was accepted for publication in *Computers & Education*. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version will be published in Spring/Summer 2008.

teachers found it “difficult and sometimes intimidating to participate as equal contributors in a technology-based development process” and suggest that effective management of collaboration can address this problem. In our case, this was not such a problem as from the outset we explicitly acknowledged teachers’ design knowledge. The success of this approach may have been because at least four partners in the project have a long history of working with particular schools and teachers, thus placing these teachers at the centre of the process. Furthermore, as distinct from DiSessa’s four models, we identified a more complex emerging structure, that of a *development network*, where distributed groups with local expertise use a pattern language to share their expertise, sometimes in collaborative long-term projects, sometimes in ad-hoc exchanges. A detailed analysis of this model is currently under development by the authors. What is becoming clear at this early stage is that a successful model needs to empower all partners in the design process, avoiding ‘producer-consumer’ and ‘sage-laymen’ relationships.

5.3 Reflections on achieving interdisciplinary design

For researchers working in TEL, gaining the experience of working in an interdisciplinary manner during the design process is essential. The ‘learning patterns’ project provided just such an opportunity. On the whole, we take a humble stance on our level of interdisciplinary work. From a starting point of not having worked together before, the first challenge was to come to an understanding of different participants’ take on design. We all engaged in it at some level: teachers with lesson planning, educational researchers with experimental design and technologists with software engineering. The first lesson learnt during the overall process is to begin from that starting point that “everyone was an expert”, and no single perspective was privileged. Reflecting back on Alexander, we see this as an extension of his work. Alexander set out to devolve design: to create a process in which “a process in which the owner is intimately involved in the evolutionary design and construction of his own house” (Alexander et al, 1973, online). Yet this process still acknowledges a hierarchy of expert and layman. The expert architect guides and enables the home-owner to design her home. By shifting to our view, we promote interdisciplinarity from the outset, where all contributors have an equal standing. As an example, educational researchers from project partner (University removed for blind review), developed the pattern ‘Own productions’ to motivate students to ask questions and state problems, rather than simply giving answers. Through working with computer scientists from partner (Institute removed for blind review) this pattern was iterated and unpacked from an initially vague principle in to smaller, concrete design elements encapsulated by the patterns ‘production’, ‘problem posing’and ‘closeness to objects’. These were further related to the existing ‘Challenge Exchange’ pattern.

The second lesson learnt during the project, was that the design knowledge we needed to share around high-level process was not all that fundamentally different. For example, for each community the high-level process of doing iterative design to better suit user needs was common. Such points of commonality proved a good grounding point for bootstrapping the interdisciplinary process, reflected early on through negotiation around the typologies.

Thirdly, as a pan-European group, scattered across six countries, the design of the web tools we required for developing patterns proved critically important. They evolved in response to our need for the interdisciplinary sharing of design knowledge at each stage of the process. They proved particularly useful for the collaborative development of the pattern language centred around the co-construction of design knowledge. As detailed in Section 3.4, the connected views provided by the pattern browser were particular helpful for discussing and critiquing the relationships between individual patterns.

Finally, moving away from our own interdisciplinary practices and on to how this interdisciplinarity was exemplified in the patterns themselves, we found that about half of the patterns in beta or release state addressed this topic. In particular, patterns discussing processes to get participants to engage with one another were evident, including 'Storyboarding', 'First Boundary Prototype' and 'Design Exploration through Gameplay' (see: http://lp.noe-kaleidoscope.org/outcomes/patterns/map/?view=flat).

In terms of defining the overall success of the IDR methodology, the production of our evolving pattern language, the development of the workshop model, and the insights gained into the relationship between educational context and the social configuration of TEL teams are concrete exemplifications of our research. Given the nature of interdisciplinary research and the difficulty in predicting their precise success factors (Sommerville & Rapport, 2000), we do not feel we are in a position to say how well our methodology will work for others. We take our position from Caruso & Rhoten (2001) that defining success "requires tenacity and a tolerance for ambiguity that many traditional researchers find difficult to maintain".

6 Conclusion

As Palmer (2002) notes "the conduct of research is […] changing. Increasingly researchers are importing and exporting information, techniques, and tools across disciplinary boundaries and working together to apply more powerful and sophisticated approaches to the questions they ask". The field of TEL is no exception, raising the difficult challenge of working across intellectual and professional boundaries. Thus, interdisciplinary design is fundamental to those involved in TEL research and practice. As noted at the beginning of this paper, there is currently a methodological weakness in current approaches to TEL design, namely that they are primarily multidisciplinary in scope. Within the 'learning patterns' project, we began to address this issue. In retrospect, we began from a good starting point, as there was a consensus on the common problem we were attempting to address. This led us to dive into the process of asking complex questions to establish a means of understanding. The IDR methodology was an enabler of generalisable design practices across the partners, through the participatory development of design patterns. We emphasize the need for practitioners and technologists to come together to construct patterns, bridging the gap between educational and technological design. However, we fully realize that (i) gaining an in-depth understanding of and (ii) applying our methodology is a non-trivial task. Undertaking participatory design in a truly interdisciplinary manner is difficult for all involved, even at the best of times. Indeed, the design process of artifacts that is undertaken independent of their pedagogy will inevitably not provide the optimal functionality. Likewise pedagogies that are designed independent of artifacts will not utilize their maximum potential. Consequently, the design of TEL environments should be done by the *community as a whole*, in an equal partnership. Following the philosophical foundations of pattern languages, we believe that success rests with empowering those who are directly involved. In contrast with other initiatives which set out to compile and distribute a set of expert-knowledge patterns, our focus is on 'pattern languages' in the sense of a constantly *negotiated norm of communication*. Thus, they originate within a community, and are continuously refined and redefined by it. In the longer term, this means using the patterns language to generate TEL environment designs, adapted to local contexts. While we do not address this point in this paper, we believe that involvement in the practice of pattern development will provide constructive insights into how best to use *any* pattern language.

Alexander stated that the moral aspect was essential to his work on patterns: they should afford the production of artifacts that are beneficial to society. This question also pertains to TEL. We have the overarching aim of improving student learning through our tools. There is a value aspect to what we do. A contribution of the IDR methodology is that it implicitly works from this standpoint, leveraging as much design knowledge from empirical experience as the participants can provide. In essence, in order to build the design process in the most productive way, there is an obligation on participants to communicate their expertise. Teachers need to stress their empirical experience from the classroom and the deep technical knowledge that software developers possess must be communicated in such a way as the functionalities of the technology are used in the best way possible for learning.

One of the reasons why TEL design is complex is because it requires coherence both in process and in outputs. By this we mean that if an interdisciplinary team coalesces, the iterative prototypes they produce will, from an early stage, reflect their design practice. Evaluation of these prototypes will then, by design, feed into the team’s interdisciplinary approach, further enhancing it. This would not be the case if only one aspect was being evaluated from a disciplinary perspective. Although interdisciplinarity remains a significant challenge for the TEL field, a contribution of the IDR methodology is to support interdisciplinary practice when designing systems that will have a lasting impact on how people learn, work and communicate. Furthermore, addressing the specific details of this challenge means undertaking system-wide thinking about the design process by understanding the ways in which patterns become a pattern language.

Our pattern language offers a modest contribution to the growing cannon of languages in various sub-fields of TEL. Yet our methodology, and associated tools, suggest an asset which could empower any interdisciplinary design community. From that perspective, our language is both a practical resource and a proof of concept. In both capacities, we see two challenges before us. On one hand, we intend to complete the cycle by deriving tools from our language and to update it based on their success. On the other hand, we wish to observe our methodology as it is adopted and adapted by new communities.

In conclusion, we return to Simon's statement from the top of this paper: "*everyone designs who devises courses of action aimed at changing existing situations into desired ones*" (Simon, 1969, p 129). We are all designers, sometimes implicitly, sometimes unaware. In order to achieve effective change, we need to develop languages which allow us to share and build on each others' design knowledge. Pattern languages have long been heralded as a powerful paradigm in this respect. However, this power does not come cheap. We need carefully designed interdisciplinary methodologies to enable us to effectively develop and use pattern languages. IDR is a step in this direction.

Acknowledgements

We would like to thank all the members of the *Learning patterns* project team for their important contributions to this research: Efi Alexopoulou (Educational Technology Lab; University of Athens); Staffan Björk (Göteborg University and the Interactive Institute, Sweden); James Bligh (CRITE, Trinity College Dublin, Ireland); Mark Childs (Institute of Education, University of Warwick, UK); Michele Cerulli (Instituto Per Le Tecnologie Didattiche, Italy); Vincent Jonker (Freudenthal Institute, The Netherlands); Chronis Kynigos (Educational Technology Lab, University of Athens); Fionnuala O' Donnell (CRITE, Trinity College Dublin, Ireland); Dave Pratt (Institute of Education, University of London); Brendan Tangney (CRITE, Trinity College Dublin, Ireland); Monica Wijers (Freudenthal Institute, The Netherlands). Thanks also to Richard Noss, Martin Oliver and Brock Craft of the London Knowledge Lab for their helpful comments on this paper. The *Learning patterns for the design and deployment of mathematical games* project was supported by the Kaleidoscope Network of Excellence, under the EC FP6 Framework.

References

Ainsworth, S. E.; Bibby, P. & Wood, D. (2002), Examining the effects of different multiple representational systems in learning primary mathematics, *Journal of the Learning Sciences*, 11(1), 25-62.

Alexander, C. (1979) *The Timeless Way of Building.* New York: Oxford University Press.

Alexander, C., Silverstein, M. and Ishikawa, S. (1977) *A Pattern Language: Towns, Buildings, Construction (Center for Environmental Structure Series).* New York, NY: Oxford University Press.

Alexander, C., Cox, M., Abdelhalim, H., Hazzard, E., Kural, I. and Schukert, M. (1973), The Grass Roots Housing Process,

http://patternlanguage.com/archives/grassroots/grassroots.htm, last retrieved 5 September 2007.

Artigue, M. (2003), Learning Mathematics in a CAS Environment: The Genesis of a Reflection about Instrumentation and the Dialectics between Technical and Conceptual Work, *International Journal of Computers for Mathematical Learning* **7**(3), 245-274.

Baggetun, R., Rusman, E., & Poggi, C. (2004). Design patterns for collaborative learning: From practice to theory and back. In L. Cantoni & C. McLoughlin (Eds.), Proceedings of the World Conference on Educational multimedia, hypermedia and telecommunications, AACE, Lugano, Switzerland (pp. 2493 – 2498).

Bannon, L. (2002) *Interdisciplinarity or Interdisciplinary Theory in CSCW?* Paper presented at the Workshop on Interdisciplinary Theory for CSCW Design, Toronto, Canada. Retrieved August 25, 2007 from: http://www.ul.ie/~idc/library/papersreports/LiamBannon/3/CSCW92f.html

Barab, S. and Squire, K. (2004) Design-Based Research: Putting a Stake in the Ground, *Journal of the Learning Sciences*, 13 (1), 1—14.

Balacheff, N. and Kaput, J. J. (1996) Computer-Based Learning Environments in Mathematics. In A.J. Bishop (Ed), *International Handbook of Mathematics Education*, (pp. 469—504). Berlin: Springer.

Bergin, J. (2000) *Fourteen Pedagogical Patterns*. Paper presented at the Fifth European Conference on Pattern Languages of Programs, Irsee, Germany. Retrieved June 11, 2006 from: http://csis.pace.edu/~bergin/PedPat1.3.html.

Bjork, S. anf Holopainen, J. (2004) *Patterns in Game Design.* Massachusetts: Charles River Media

Borchers, J. O. (2001), A Pattern Approach to Interaction Design, AI & Society Journal of Human-Centred Systems and Machine Intelligence 15(4), 359-376.

Brown, A. L. (1992) Design Experiments: Theoretical and Methodological Challenges in Creating Complex Interventions in Classroom Settings. *Journal Of the Learning Sciences*, 2 (2)**,** 141—178.

Bryant, A. (2002) Grounding Systems Research: Re-establishing Grounded Theory, Proceedings of the Hawaii International Conference on System Sciences, pp. 253.

Committee on Science, Engineering and Public Policy (2004) *Facilitating Interdisciplinary Research*. Washington D.C.: National Academic Press

Caruso D. and Rhoten D. (2001) *Lead, Follow, Get out of the Way: Sidestepping the Barriers to Effective Practice of Interdisciplinarity*. http://www.hybridvigor.net/interdis/pubs/hv_pub_interdis-2001.04.30.pdf, last visited 28 August 2007.

Dearden, A., Mcmanus, B., Allgar, E. and Finlay, J. (2002a) *Using Pattern Languages in Participatory Design*. Paper presented at the Computer Human Interaction Conference: Workshop on Patterns in Practice, Minneapolis, USA.

Derntl. M. and Motschnig-Pitrik, R. (2005) The Role of Structure, Patterns, and People in Blended Learning. *The Internet and Higher Education*, 8 (2), 111—130.

DiSessa, A. A.; Azevedo, F. S. & Parnafes, O. (2004), 'Issues in Component Computing: A Synthetic Review', Interactive Learning Environments 12(1), 109-159.

Disessa, A. A. and Cobb, P. (2004) Ontological Innovation and the Role of Theory in Design Experiments. *Journal of the Learning Sciences,* 13 (1)**,** 77–103.

Erickson, T. (2000) Lingua Franca for design: sacred places and pattern languages. In D. Boyarski and W. Kellogg (Eds) *Proceedings of the conference on Designing interactive systems*,(pp. 357—368). New York, USA: ACM Press.

Engeström, Y. (1987) Perspectives on Activity Theory (Learning in Doing: Social, Cognitive and Computational Perspectives), Cambridge: Cambridge University Press.

Friedrich, W.R. and Van Der Poll, J.A. (2007) Towards a Methodology to Elicit Tacit Domain Knowledge from Users. In *Proceedings of the 2007 Informing Science and IT Education Joint Conference*, Ljubljana, Slovenia

Gamma, E., Vlissides, J., Johnson, R. and Helm, R. (1995) *Design Patterns,* Boston, MA: Addison-Wesley

German, D. M. and Cowan, D. D. (2000) Towards a unified catalog of hypermedia design patterns. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences* (pp. 6067—6073) . Washington DC, USA: IEEE Computer Society.

Gibbons, M.; Limoges, C.; Nowotny, H.; Schwartzman, S.; Scott, P. & Trow, M. (1994), The New Production of Knowledge: The dynamics of science and research in contemporary societies, London: Sage Publications.

Goodyear, P. (2005) Educational design and networked learning: Patterns, pattern languages and design practice, Australasian Journal of Educational Technology, 21(1), 82-101.

Guin, D. & Trouche, L. (1999), The complex process of converting tools into mathematical instruments: the case of calculators, *International Journal of Computers for Mathematical Learning* 3(3), 195--227.

Harrison, N. (1999),The Language of Shepherds: A Pattern Language for Shepherding, in 'Proceedings of the 6th Annual Conference on the Pattern Languages of Programs', pp. 15-18.

Kafai, Y. B. (1995), *Minds in Play: Computer Game Design As a Context for Children's Learning*, New York: Lawrence Erlbaum Associates.

Kali. Y (2006) Collaborative knowledge building using a design principles database. *International Journal of Computer Supported Collaborative Learning*, (1) 2.

Kaptelinin, V. and Nardi, B. (2006). Acting with Technology: Activity Theory and Interaction Design. Cambridge: MIT Press.

Kreimeier, B. (2002). The case for game design patterns. Available from http://www.gamasutra.com/features/20020313/kreimeier_03.htm

Kynigos C. (2002). Generating Cultures for Mathematical Microworld Development in a Multi-Organisational Context. Journal of Educational Computing Research, (1 & 2), 183-209.

Noss, R. and Hoyles, C. (1996) *Windows on Mathematical Meanings : Learning Cultures* *and Computers.* Dordrecht: Kluwer.

Mor, Y. and Winters, N. (2007) Design approaches in technology enhanced learning, *Interactive Learning Environments* 15(1), 61-75

Mor, Y., Winters, N., Pratt, D. and Bjork, S. (2007) Tools for developing design patterns for mathematical computer games. In *Proceedings of Games-in-Action*, Gotenburg, Sweden.

Palmer, C.L. (2002) *Working at the Boundaries of Science: Information and the Interdisciplinary Research Process*. Berlin: Springer

Popper, K. R. (1963) Conjectures and Refutations: The Growth of Scientific Knowledge. New York: Routledge and Kegan Paul.

Quintana, C., Reiser, B., Davis, E,. Krajcik, J., Golan, R., Kyza, E.,Edelson, D. and Soloway, E. (2002) Evolving a Scaffolding Design Framework for Designing Educational Software, Proceedings of International Conference of the Learning Sciences, 359-366.

Rabardel, P. (2003) From artefact to instrument, Interacting with Computers 15(5), 641-645.

Radford, L. (2000) Signs and meanings in students' emergent algebraic thinking: a semiotic analysis. *Educational Studies in Mathematics*, 42 (3), 237—268.

Retalis, S. Georgiakakis, P. & Dimitriadis, Y. (2006), 'Eliciting design patterns for e-learning systems', Computer Science Education 16(2), 105--118.

Scaife, M., Rogers, Y., Aldrich, F. and Davies, M. (1997) Designing for or designing with? Informant design for interactive learning environments In Proceedings of CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM,, pp. 343-350. http://dx.doi.org/10.1145/258549.258789

Schön, D. A. (1983) The Reflective Practitioner: How professionals think in action. London: Temple Smith

Schön, D. A., & Bennett, J. (1996). Reflective conversation with materials. In T. Winograd (Ed.), Bringing design to software pp. 171-184. New York: Addison Wesley.

Simon, H. A. (1969) *The Sciences of the Artificial,* Cambridge, MA: The MIT Press

Soloway, E., Guzdial, M. and Hay, K. (1994). Learner-Centered Design: The Challenge for HCI in the 21st Century. Interactions 1(2). pp 36-47.

Somerville, M. and David R., (2000) *Transdisciplinarity: reCreating Integrated*

Knowledge. Oxford: EOLSS Publishers Co. Ltd.

Star, S. L. (1990), 'The structure of ill-structured solutions: boundary objects and heterogeneous distributed problem solving', *Morgan Kaufmann Series In Research Notes In Artificial Intelligence*, 37--54.

Star, S. L. & Griesemer, J. R. (1989), 'Institutional Ecology,Translations' and Boundary

Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39', *Social Studies of Science* **19**(3), 387

van den Besselaar, P. & Heimeriks, G. (2001),'Disciplinary, Multidisciplinary, Interdisciplinary - Concepts and Indicators', Paper for the 8th conference on Scientometrics and Informetrics, Sydney. Australia

Wastell, D. G. (2001). Barriers to effective knowledge management: Action Research Meets Grounded Theory, Proceedings of 9th European Conference on Information Systems.

Winters, N. and Mor, Y. (2007) Participatory design in open education: a workshop model for developing a patterns language, Proceedings of OpenLearn 2007, Milton Keynes, UK